



Apache Kafka Training Course: Real-Time Data Streaming, Kafka Connect and CDC



Apache Kafka Training Course: Real-Time Data Streaming, Kafka Connect and CDC

Introduction:

Apache Kafka training covering real-time data streaming, Kafka Connect and CDC is a 5-day course for data engineers, back-end developers and platform engineers that ends with a CDC-to-Analytics Streaming Pipeline and Operations Runbook. Organisations still move operational data in nightly batches or through fragile point-to-point integrations, so fraud signals, stock levels and customer events arrive hours late and every new consumer needs another custom feed. Nominees already write code or run data platforms at work and spend each day in hands-on labs on a running Kafka cluster. CoreConcept Training Center delivers this Apache Kafka and real-time data streaming course.

Course Objectives:

- Design topic, partition and replication layouts on a KRaft-based Kafka cluster that meet ordering, durability and throughput targets
- Configure producers and consumer groups for at-least-once or exactly-once delivery using idempotence, transactions and offset commit control
- Capture row-level database changes with Debezium source connectors on Kafka Connect and route them to topics and sink systems
- Govern event formats with a schema registry, Avro or Protobuf serialisers and compatibility rules that let producers evolve safely
- Build stateful stream processing jobs in Kafka Streams and Apache Flink with event-time windows, watermarks and stream-table joins
- Secure, monitor and tune a streaming platform with TLS, SASL, ACLs, consumer lag alerts and partition rebalancing

Target Audience:

- Data engineers responsible for ingestion pipelines that must deliver operational changes within seconds rather than overnight
 - Back-end developers responsible for services that publish or consume business events across teams
 - Platform and site reliability engineers responsible for running, securing and scaling Kafka clusters
 - Integration engineers responsible for replacing point-to-point feeds with shared event topics
 - Analytics engineers responsible for feeding dashboards, search indexes and feature stores from live data
-

Course Outline:

Day 1: Event-Driven Architecture and Kafka Log Foundations

- Event-Driven Architecture Versus Request-Response Integration Trade-Offs
- Kafka Commit Log Model with Topics, Partitions and Offsets
- Broker Roles and KRaft Controller Quorum Metadata Management
- Replication Factor, In-Sync Replicas and Leader Election Behaviour
- Streaming Use Case Inventory and Latency Requirement Mapping

Day 2: Producers, Consumer Groups and Delivery Semantics

- Producer Batching, Linger Time and Compression Codec Choices
- Partition Key Design, Ordering Guarantees and Hot Partition Risks
- Consumer Group Rebalancing, Cooperative Assignment and Static Membership
- Offset Commit Strategies for At-Most-Once and At-Least-Once Processing
- Idempotent Producers and Kafka Transactions for Exactly-Once Delivery

Day 3: Kafka Connect, Debezium Change Data Capture and Schema Management

- Kafka Connect Distributed Workers, Tasks and Converter Configuration
- Debezium Source Connectors Reading Database Transaction Logs
- Single Message Transforms and Topic Routing for Change Events
- Schema Registry Subjects with Avro and Protobuf Serialisers
- Backward and Forward Compatibility Rules for Event Schema Evolution

Day 4: Stream Processing, Security and Cluster Operations

- Kafka Streams KStream, KTable and State Store Topologies
- Apache Flink Event Time, Watermarks and Late Event Handling
- Tumbling, Hopping and Session Windows with Stream-Table Joins
- TLS Encryption, SASL Authentication and ACL Authorisation Rules
- Consumer Lag Alerts, JMX Metrics and Partition Reassignment Tuning

Day 5: Lab Build of a CDC-to-Analytics Streaming Pipeline

- Managed Cloud Kafka Services Versus Self-Managed Cluster Selection
 - Lab Debezium Capture of an Order Database into Topics
 - Lab Flink Windowed Aggregation Feeding an Analytics Store
 - Lab Failure Drills for Broker Loss and Consumer Rebalance
 - CDC-to-Analytics Streaming Pipeline and Operations Runbook Completion
-

Skills You Will Gain:

- Event Streaming Architecture
- Topic and Partition Design
- Delivery Guarantee Configuration
- Change Data Capture Engineering
- Schema Evolution Governance
- Stateful Stream Processing
- Kafka Cluster Hardening
- Streaming Platform Observability

Why Attend This Course:

- Deliver a CDC-to-Analytics Streaming Pipeline and Operations Runbook to the data platform lead as a reference for the next streaming use case
- Decide partition counts, replication settings, delivery guarantees and whether Kafka Streams or Apache Flink suits a given processing job
- Avoid duplicated or lost events, breaking schema changes, runaway consumer lag and unsecured topics before they reach production
- Coach colleagues on connector configuration, schema compatibility checks and stream processing patterns tested in the labs

Conclusion:

Back at work, the participant hands the CDC-to-Analytics Streaming Pipeline and Operations Runbook to the data platform lead and the owners of the source databases. The team uses it to agree topic naming, retention, schema compatibility and security settings, and to approve which operational tables are streamed next and which consumers may read them. After the pipeline has run under real load for a few weeks, the unit should review end-to-end latency, consumer lag peaks, connector restarts, rejected schema changes and broker disk growth, and retune partitions and retention where results diverge from targets.

Frequently Asked Questions FAQ:

What should participants know before the Apache Kafka real-time data streaming course?

Participants should already write code in Java, Python or another language, use SQL and the Linux command line, and understand basic networking. A laptop able to run containers is needed for the labs, where a Kafka cluster, a source database and processing jobs run locally.

How does the Apache Kafka real-time data streaming course differ from a batch ETL or big data analytics course?

It works on events as they happen: topics, delivery guarantees, change data capture, stateful stream processing and cluster operations. Batch ETL and warehouse courses load data on a schedule, and big data analytics courses focus on querying large stored datasets rather than running a live event platform.

Why do teams pair Apache Kafka with change data capture for real-time data streaming?

Change data capture reads committed changes from a database transaction log and publishes them as events, so other systems receive inserts, updates and deletes within seconds without polling queries or changes to the source application.

What do participants take back from the Apache Kafka real-time data streaming course?

Participants take back a working CDC-to-Analytics Streaming Pipeline with its connector, schema and stream processing configuration, plus an Operations Runbook covering security settings, monitoring alerts, lag thresholds and failure recovery steps.